INSH 6406, Fall 2020
Analyzing Complex Digitized Data

Instructor

Laura K. Nelson
Email: l.nelson@northeastern.edu
Twitter: @LauraK_Nelson
Office hours: Thursdays, 11am-1pm, online

Course Description

Meetings: Tuesdays, 5:30pm – 8:50pm, synchronous online and asynchronous on Canvas

Readings:

We will be using one main textbook:

Charles R. Severance. Python for Everybody: Exploring Data Using Python 3. This book is free and open source (open science, it's a thing!). There are multiple online options, or you can buy the book on the cheap if you prefer a hard copy.

Other readings are available online through the library or, in select cases, on Canvas. If you are having problems accessing any of the readings let me know.

Overview:

Data are everywhere y'all. Historical documents, literature and poems, diaries, political speeches, government documents, emails, text messages, social media, images, maps, cell phones, wearable sensors, parking meters, credit card transactions, Zoom, surveillance cameras. You get it. You know it. Combined with rapidly expanding computational power and increasingly sophisticated algorithms, we are where we are. Privacy, ethics, surveillance, bias, discrimination, yes. But also incredible potential for better understanding the social world, and the potential to use data for good.

In this course we will explore how data, digital material and digital methods are impacting the humanities and social sciences. We'll also reflect critically on how these materials and methods are being used inside and outside of academia.

We'll explore these issues from three perspectives: the technical skills necessary to access and analyze data (Python and computers!), the epistemology underlying these methods and best practices re: research design (hint: there is not just one way to do computational research), and the practical knowledge we and others can produce using digital data and methods.

Learning Goals

This course is NOT a computer science course. It is not even a course on computer programming. No prior programming experience is required or assumed. It is primarily a social science and humanities course, with an eye toward digital technologies. We will not have computers analyze data or cultural material for us. Instead, we will harness the superior ability for computers to count and extract patterns from complex data and cultural material, and use this output to enhance our own critical thinking and interpretive analyses. To implement these methods we will use the open source (and free!) programming language Python and the Jupyter platform.

Specific skills covered include collecting digitized data, structuring digitized data, data formats, and an introduction to text and network analysis – with a smudge of machine learning. The ultimate goal is to encourage you to think about novel and creative ways you can apply these techniques to your own area of study.

By the end of the course you will have a better understanding of the range of types of digital data available, different ways of collecting and structuring them, ways computers can help you answer questions, what kind of evidence the different techniques produce, and how this evidence can be used to help you better understand the social world.

Learning Outcomes

By the end of the course you should be able to:

1. Explain three different ways computers are being used in social science and humanities research to ask and answer questions
2. Know enough Python basics to qualify as, at a minimum, a novice programmer
3. List three different types of digital data (e.g., delimited separated files, raw text, json), be able to write Python code to input and process each type, and explain how and why you might use each data type in research
4. Write Python code to collect and structure digitized data, including from APIs, process the data, and produce two or three visualizations and/or output to explore or analyze the data
5. Explain what the output from computational methods means, and derive a few insights about the social world from the output and visualizations
6. Feel comfortable learning new techniques and new Python libraries on your own

Course Format

The course will be split into four main parts: basics of Python, text analysis, network analysis and visualization, and the basics of machine learning. The goal is for you to be able to use these different types of data and approaches to explore research applications of your choosing. As such, you will complete a short research project for the last three parts: text analysis, network analysis/visualization and machine learning.

The form of these three projects will be a Jupyter notebook, also called a computational essay. A Jupyter notebook is an interactive computational environment that allows you to combine text, code, output, and visualizations into one document, and easily share the document with colleagues or publish it on the web. It can be used with a variety of programming languages, including Python. Because it is a functioning program environment that also can incorporate text and visualizations in a seamless and visually pleasing manner, it is popularly used to teach programming and computational methods, to present scientific findings, and it is starting to be widely used in industry, including data-driven journalism. As such, your completed final project will be an excellent addition to your resume or CV. Here's an example of a Jupyter notebook, and a good one to emulate for your own projects:

http://nbviewer.jupyter.org/github/brianckeegan/Bechdel/blob/master/Bechdel_test.ipynb

Course Requirements

Technology Requirements

Students must have access to a laptop or desktop, and you must have access to it during class every day. If you do not have a laptop contact me and we can try to work something out.

This workshop will be taught in the open source programming language Python and the programming environment Jupyter. Participants should install Anaconda prior to the first day of class (try your best, be we'll trouble shoot that first day if you have difficulties):

- Anaconda for Python 3.8 (https://www.anaconda.com/download/). Anaconda includes Python, a Python interpreter, the necessary Python packages, and Jupyter. I recommend using the Graphical Installer.

Grading and Assignments

10%	Attendance and participation - participation in class discussions, synchronously or asynchronously - completing class tutorials, synchronously or asynchronously
20%	Weekly reading responses (3 points each, up to 21 points)
20%	Weekly programming exercises (3 points each, up to 21 points)
50%	Three research projects and presentations

Course Structure

To facilitate learning both programming and domain knowledge, the course will consist of practical tutorials aimed at getting you processing and analyzing cultural material via Python, as well as discussions about assigned readings that explore a practical question, epistemology, or issue using computational techniques. It is important that you complete the readings before each class and come prepared to discuss the material. During these discussions there will be space to critique the material and these methods. It is important that we respect one another's thoughts, give everybody the space to talk, and address our comments at the ideas and not the person.

The course will meet virtually once a week for 3.2 hours. Class time will be a combination of lecture, discussion, hands-on tutorials, and programming practice.

Class time will be structured as follows:

5:30 - 6:00: programming tutorial
6:00 - 6:30: in-class exercises (cameras off)
6:30 - 6:45: exercise solutions
6:45 - 7:00: break (cameras off)
7:00 - 7:15: introduce substantive themes
7:15 - 7:30: written reflections (cameras off)
7:30 - 8:15: substantive discussion
8:15 - 8:30: break (cameras off)
8:30 - 8:50: practicum – tutoring/get started on your programming exercise

The class will be synchronous, but I understand that there may be reasons why folk can't show up every week. We will use online tools, including collaborative class notes, so that everyone can participate each week regardless of whether they can show up to class. Your grade will be based on class participation (synchronous or asynchronous), reading responses, programming exercises, and three research projects.

Assignments

Reading Responses must be submitted by 10PM on Mondays. Please read everyone's reading response before class on Tuesday. The programming exercises must be submitted by 10PM on Sunday.

In lieu of a final exam, you will explore questions relevant to the humanities and/or social sciences in three research projects using three different types of digitized data or material and computational techniques: text, networks, and machine learning. These projects are designed to encourage you to creatively combine the knowledge and skills built through the semester to explore questions about the social world. I will hand out detailed rubrics closer to the due date for each project. We will set aside some class time to view and comment on each other's projects.

Attendance and Participation

Each week we will learn skills and develop knowledge that builds on previous skills learned, so it is important to attend every class (there will be options to participate asynchronously as needed). Learning Python is like learning a foreign language. The best way to learn it is to use it all the time. To encourage the continual use of the skills you are learning we will complete short exercises during class (or asynchronously as needed).

Reading Responses

In this course humanities and social science questions are central, and computational techniques are used to answer those questions. There will be four types of assigned readings: (1) a Python textbook to help you get familiar with computer science basics; (2) articles exploring the epistemology underlying computational analyses; (3) articles that introduce broad categories of computational methods (e.g., text analysis, network analysis, machine learning), and (4) articles that apply computational methods to answer questions about the social world. You will have 12 opportunities during the semester to submit a short (two-paragraphs to one-page) reading response. You can get up to 3 points per reading response, up to 21 points. Once you reach 21 points, you're done! The responses are due by 10pm the day before class. These short responses will help you understand and evaluate the applied use of these methods, they will help you get used to writing and talking about computational methods and the different types of evidence these methods produce, and they will help frame the class discussion. Everyone should read all of the reading responses prior to class. I will kick off the substantive discussion by posting themes from the responses in a collaborative doc, and then will invite everyone to comment/respond to these themes. That will guide our subsequent discussion, and will provide running notes for the class.

Programming Exercises

As with learning any foreign language, the best way to learn Python is to keep writing code. You are required to complete (short) weekly programming exercises, designed to give you practice writing code. These will also help me gauge the speed at which we're going through the programming material. You can get up to 3 points for each programming exercise for a total of 21 points. While I encourage you to do the programming exercises every week, once you hit 21 points you're done. The programming exercises are due by 10pm on Sunday.

Research Projects

The goal of the research projects is to creatively combine the techniques you learned in the course to explore questions related to the humanities or social sciences. Note that this is an introductory course, and I will present a lot of material throughout the semester, so these projects will by necessity be only preliminary explorations of research questions. The goal is to provide you enough technical skills in the course so you can explore substantive questions further on your own, through your own projects or in other classes.

Through these projects you should show that you understand (a) what types of questions are interesting or important to humanists and/or social scientists, (b) what types of questions can be best answered using computational or digital techniques, (c) what types of techniques and evidence are appropriate to best answer your question, and (d) that you can think about how to present your findings and analysis in a reproducible way and in a way that supports, and persuades others of, your conclusion.

Keeping the above goals in mind, your research projects Jupyter notebooks should include the following:

1. 1-2 cells describing the question or puzzle you are exploring, why it is interesting or important, how others have attempted to answer this question, and how you are improving on these answers. If no one has addressed this question, explain why you think this is the case. In other words, what are you doing that's different than what others have done?

2. 2-4 cells describing the data or material you are using to explore the question and how you collected the data or material. These cells should include summary statistics of the data/material. If appropriate, describe what your data or material are representative of.

3. 2-10 cells containing the analysis or steps toward an analysis. These cells should contain a description of the planned analysis process and why it is appropriate for your question and data/material, followed by code implementing either some of the techniques or at least provides some summary descriptions of your data or material, the output from the calculations or the summary descriptions of your data or material, and a description of how you understand the output.

4. 1-2 cells producing some sort of data visualization or data summary output.

5. 1-2 cells detailing your interpretation of the output, and broader conclusions about history and/or the world around you that you draw from your exploration, or that you would hope to draw if you carried the project further. Support your interpretation with evidence from your analysis. End with suggestions for further analyses and other data or material that could help us continue to explore your question.

Questions? Discussion Board, Office Hours, and Email

If you come across errors as you run code that you can not solve, post them to the discussion board on Canvas (start a new thread for new errors). You may also post questions or comments about the readings or about your research projects. I encourage everyone to answer each other's questions, as this is the best way to learn complicated material. Often many people will get the same error or will have similar questions, so check the discussion board for answers before posting your error or question. This is not the comments section on YouTube, so keep your comments respectful. Disrespect will absolutely not be tolerated.

You are also encouraged to come to my virtual office hours. Email should only be used for quick logistical questions or if you need to inform me of a planned absence. I will get back to emails within 16 working-hours, so plan ahead. My general philosophy is to work hard during the week, and to take weekends off. If you email me or post questions on a Friday afternoon or a weekend, I may not respond until the following Monday.

Consulting Resources

I encourage you to take advantage of the Digital Scholarship Group at Northeastern. They offer a wealth of services – including digital data collections – and can offer advise on collecting and structuring digital data. They also offer a quiet space to work.

Note on Plagiarism

I encourage you to work together to help each other review the readings and to learn the coding. However, *all written and coding work must be your own*. I take academic honesty seriously, and you should too.

This class has very strict standards for borrowing code: if you borrow anything for use in your projects, you must have a citation. A good guideline is that if you take more than three lines of code from some source, you must include the information on where it came from. A URL or a notation (e.g., "Pandas help files") is fine. If it is an entire function, note it at the beginning of the code segment and include any original credit information. Provide a qualitative description of what you used, and what you changed/contributed. If you are unsure about this policy, ask the instructor. The university's academic integrity policy discusses actions regarded as violations and consequences for students.

For more information on your rights and responsibilities as a student see:
http://www.northeastern.edu/osccr/academic-integrity

Course Schedule

Week	Date	Theme ************************ Notes	Readings
1	(no class)	------	------
2	Sep. 15	installation check, python and markdown, and course overview ************************ come to class with Anaconda 3.8 installed.	*Python for Everybody*, Chapters 1, 2, and 3 The syllabus
3	Sep. 22	introduction to computational thinking ************************	*Python for Everybody*, Chapters 5, 6, and 8 Tukufu Zuberi (2008). "Deracializing Social Statistics Problems in the Quantification of Race." Pp. 127-136 in *White Logic, White Methods: Racism and Methodology*, edited by Tukufu Zuberi and Eduardo Bonilla-Silva. Rowman & Littlefield Publishers. Sebastian Benthall (2016). "Philosophy of Computational Social Science." *Cosmos and History: The Journal of Natural and Social Philosophy* 12 (2): 13-30. Quincy Thomas Stewart (2008). "Swimming Upstream: Theory and Methodology in Race Research." Pp. 111-125 in *White Logic, White Methods: Racism and Methodology*, edited by Tukufu Zuberi and Eduardo Bonilla-Silva. Rowman & Littlefield Publishers. *Optional (just for fun)*: Book Review: What is Digital Sociology
4	Sep. 29	research strategies and data analysis ************************	*Python for Everybody*, Chapters 4, 7 Introduction to Pandas Mike Savage and Roger Burrows (2007). "The Coming Crisis of Empirical Sociology." *Sociology* 41(5): 885–899. Gurminder K Bhambra (2016). "Postcolonial Reflections on Sociology." *Sociology* 50(5) 960–966.

			Jessie Daniels and Polly Thistlethwaite (2016). “Being a scholar-activist then and now.” Pp. 21-38 in *Being a scholar in the digital era: Transforming scholarly practice for the public good*. Handout: Originating, Specifying, and Central Questions
5	Oct. 6	text analysis and counting words ************************	*Python for Everybody*, Chapters 9 and 10 Justin Grimmer and Brandon S. Stewart (2013). “Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts.” *Political Analysis*: 1-31. Christopher A. Bail, Taylor W. Brown, and Marcus Mann (2017). “Channeling Hearts and Minds: Advocacy Organizations, Cognitive-Emotional Currents, and Public Conversation.” *American Sociological Review* 82(6): 1188-1213.
6	Oct. 13	NLP using NLTK ************************	Monica Lee and John Levi Martin (2015). “Coding, counting and cultural cartography.” *American Journal of Cultural Sociology* 3: 1–33. Laura K. Nelson (2020). “Computational Grounded Theory: A Methodological Framework.” *Sociological Methods and Research*. John W. Mohr, Robin Wagner-Pacifici, and Ronald L. Breiger. 2015. “Toward a Computational Hermeneutics.” *Big Data & Society*.
7	Oct. 20	APIs and collecting data ************************	Moya Bailey (2015). “#transform(ing)DH Writing and Research: An Autoethnography of Digital Humanities and Feminist Ethics.” *Digital Humanities Quarterly* 9(2). Catherine D'Ignazio and Lauren Klein (2020). “The Power Chapter.” *Data Feminism*. Colin Jerolmack and Shamus Khan (2017). “The Analytic Lenses of Ethnography.” *Socius*. Handout: Where to find data
8	Oct. 27	social network analysis ************************	Nicholas A. Christakis and James H. Fowler (2009). “Chapter 1: In the Thick of It.” *Connected: The Surprising Power of Our Social Networks and How They Shape Our Lives*.

			Shin-Kap Han (2009). "The Other Ride Of Paul Revere: The Brokerage Role In The Making Of The American Revolution" *Mobilization: An International Quarterly* 14(2): 143-162. Sarah J. Jackson and Brooke Foucault Welles (2016). "#Ferguson is Everywhere: Initiators in Emerging Counterpublic Networks." *Information, Communication & Society* 19(3): 397-418.
9	Nov. 3	data visualization 1 ************************ **First research project (text) due before class**	Kieran Healy and James Moody (2014). "Data Visualization in Sociology." *American Review of Sociology*. 40: 105-28. Catherine D'Ignazio and Lauren Klein (2020). "On Rational, Scientific, Objective Viewpoints from Mythical, Imaginary, Impossible Standpoints." *Data Feminism*.
10	Nov. 10	data visualization 2: maps! ************************	Daniel T. O'Brien et al. (2020). "Urban Heat Islets: Street Segments, Land Surface Temperatures, and Medical Emergencies During Heat Advisories." *American Journal of Public Health* 110 (7): pp. 994-1001. Katherine McKittrick (2011). "On plantations, prisons, and a black sense of place." *Social & Cultural Geography* 12 (8): 947-963. Kevin M. Kruse (2019). "How Segregation Caused Your Traffic Jam." *New York Times Magazine*.
11	Nov. 17	machine learning 1 ************************	S. B. Kotsiantis (2007). "Supervised Machine Learning: A Review of Classification Techniques." *Informatica* 31: 249-268. Ziad Obermeyer et al. (2019). "Dissecting racial bias in an algorithm used to manage the health of populations." *Science* 366 (6464): 447-453. Ruha Benjamin (2019). "Assessing risk, automating racism." *Science* 366 (6464): 421-422
12	Nov. 24	machine learning 2 ************************ **Second research project (networks) due before class**	Rochelle Terman (2017). "Islamophobia and Media Portrayals of Muslim Women: A Computational Text Analysis of US News Coverage." *International Studies Quarterly* 61(3): 489-502. Austin C. Kozlowski, Matt Taddy, and James A. Evans (2019). "The Geometry of Culture: Analyzing Meaning through Word Embeddings." *American Sociological Review* 84(5).
13	Dec. 1	machine vision	Jackelyn Hwang and Robert J. Sampson (2014). "Divergent Pathways of Gentrification: Racial Inequality and the Social

		************************	Order of Renewal in Chicago Neighborhoods." *American Sociological Review* 79(4), 726–751. Kashmir Kill (2020). "Wrongfully Accused by an Algorithm." *New York Times,* June 24. Alex Hanna et al. (2020). "Towards a Critical Race Methodology in Algorithmic Fairness." In *Conference on Fairness, Accountability, and Transparency* (FAT* '20), January 27–30, 2020, Barcelona, Spain. ACM, New York, NY, USA
14	Dec. 8	presentations + wrap-up ************************	
15	Dec. 15	Final (no class) ************************ **Third research project (machine learning) due by 5:30pm**	------------------------